Get Shaded

- João Branquinho
- Tiago Ramalho

Why Bitcoin?

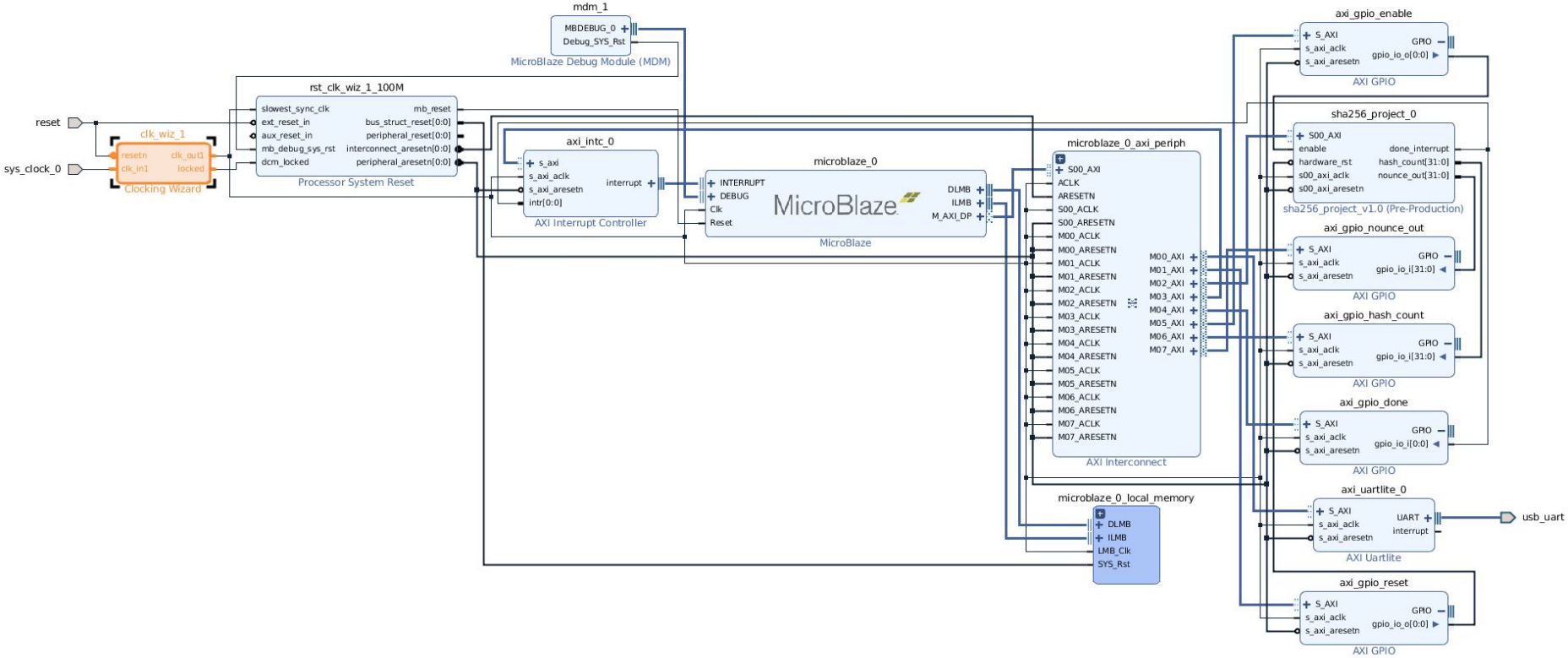
- Low Fees
- Reduced risk
- Full control over the investment
- Possible investment

Problem approach

```
para cada pedaço
    criar um array de mensagens de 64 espaços w[0..63] de palavras de 32 bits
    (Os valores iniciais em w[0..63] não importam, muitas implementações os zeram nesse ponto
    copiar o pedaço nas primeiras 16 palavras w[0..15] do array agendado de mensagens
```

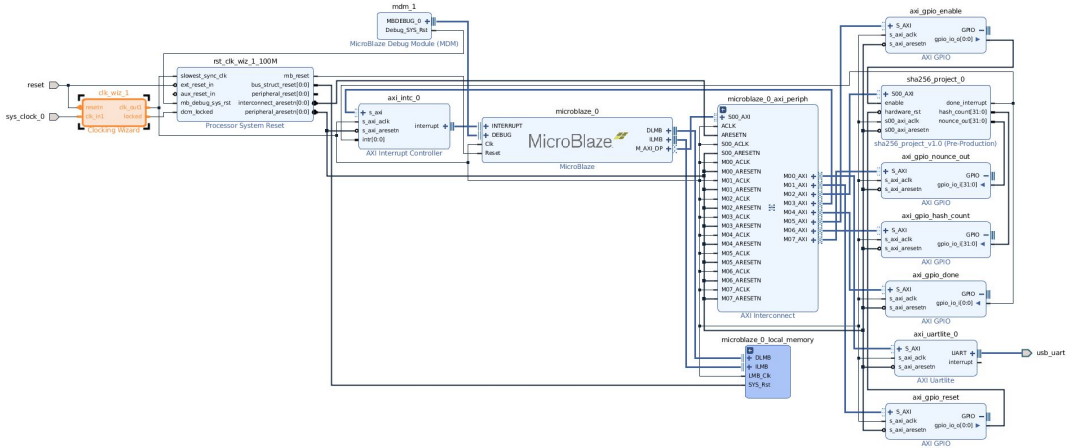
```
    Estender as primeiras palavras de 16 bits nas 48 palavras restantes w[16..63] do array:
    para i de 16 to 63
        s0 := (w[i-15] rotacionarparadireita 7) xor (w[i-15] rotacionarparadireita 18) xor (w[i-15] deslicarparadireita 3)
        s1 := (w[i-2] rotacionarparadireita 17) xor (w[i-2] rotacionarparadireita 19) xor (w[i-2] deslocarparadireita 10)
        w[i] := w[i-16] + s0 + w[i-7] + s1
```

```
Laço principal da função de compressão:
para i de 0 até 63
    S1 := (e rotacionarparadireita 6) xor (e rotacionarparadireita 11) xor (e rotacionarparadireita 25)
    ch := (e e f) xor ((não e) e g)
    temp1 := h + S1 + ch + k[i] + w[i]
    S0 := (a rotacionarparadireita 2) xor (a rotacionarparadireita 13) xor (a rotacionarparadireita 22)
    maj := (a e b) xor (a e c) xor (b e c)
    temp2 := S0 + maj
```


reset
sys_clock_0
clk_wiz_1
resetn
clk_in1
clk_out1
locked
Clocking Wizard
rst_clk_wiz_1_100M
slowest_sync_clk
ext_reset_in
aux_reset_in
mb_debug_sys_rst
dcm_locked
mb_reset
bus_struct_reset[0:0]
peripheral_reset[0:0]
interconnect_aresetn[0:0]
peripheral_aresetn[0:0]
Processor System Reset
mdm_1
MBDEBUG_0
Debug_SYS_Rst
MicroBlaze Debug Module (MDM)
axi_intc_0
s_axi
s_axi_aclk
s_axi_aresetn
intr[0:0]
interrupt
AXI Interrupt Controller
microblaze_0
INTERRUPT
DEBUG
Clk
Reset
MicroBlaze
DLMB
ILMB
M_AXI_DP
microblaze_0_axi_periph
S00_AXI
ACLK
ARESETN
S00_ACLK
S00_ARESETN
M00_ACLK
M00_ARESETN
M01_ACLK
M01_ARESETN
M02_ACLK
M02_ARESETN
M03_ACLK
M03_ARESETN
M04_ACLK
M04_ARESETN
M05_ACLK
M05_ARESETN
M06_ACLK
M06_ARESETN
M07_ACLK
M07_ARESETN
M00_AXI
M01_AXI
M02_AXI
M03_AXI
M04_AXI
M05_AXI
M06_AXI
M07_AXI
AXI Interconnect
microblaze_0_local_memory
DLMB
ILMB
LMB_Clk
SYS_Rst
axi_gpio_enable
S_AXI
s_axi_aclk
s_axi_aresetn
GPIO
gpio_io_o[0:0]
AXI GPIO
sha256_project_0
S00_AXI
enable
hardware_rst
s00_axi_aclk
s00_axi_aresetn
done_interrupt
hash_count[31:0]
nounce_out[31:0]
sha256_project_v1.0 (Pre-Production)
axi_gpio_nounce_out
gpio_io_i[31:0]
axi_gpio_hash_count
axi_gpio_done
gpio_io_i[0:0]
axi_uartlite_0
UART
interrupt
usb_uart
AXI Uartlite
axi_gpio_reset

Arquitecture

- ~ 3M block hashes per sec
  - 608 bits extended to 1024
  - 256 bits extended to 512



Challenges and Difficulties